(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
7 March 2002 (07.03.2002)

PCT

(10) International Publication Number
WO 02/19684 A2

(51) International Patent Classification[7]: H04M 11/06

(21) International Application Number: PCT/US01/41870

(22) International Filing Date: 23 August 2001 (23.08.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/230,051 1 September 2000 (01.09.2000) US
09/755,465 4 January 2001 (04.01.2001) US

(71) Applicant: CENTILLIUM COMMUNICATIONS, INC. [US/US]; 47211 Lakeview Boulevard, Fremont, CA 94538 (US).

(72) Inventors: DODDAPANENI, Ashok; Apartment 202, 555 E. El Camino Real, Sunnyvale, CA 94087 (US). MORADI, Mohammad; 1069 Yorktown Drive, Sunnyvale, CA 94087 (US).

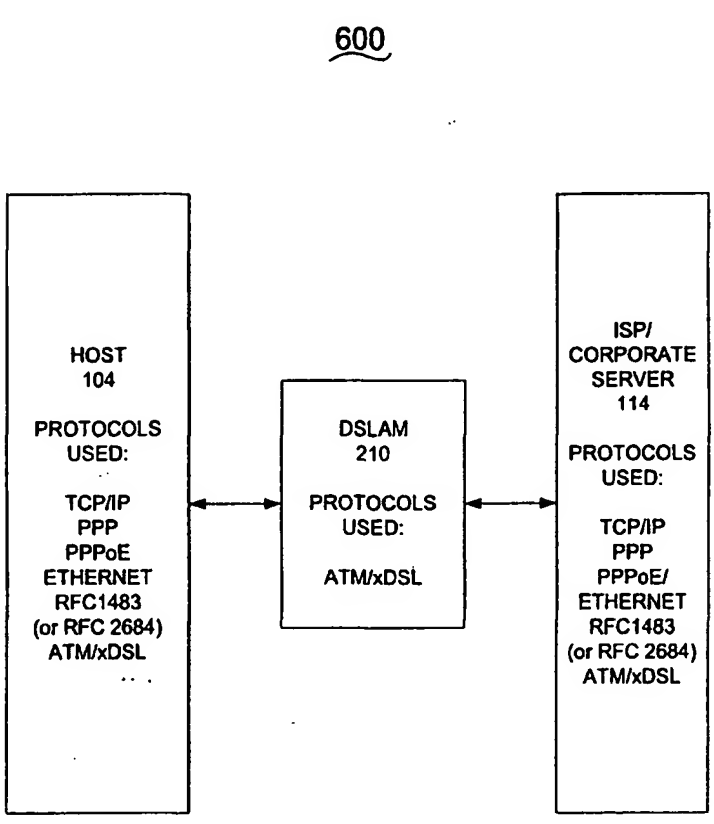
(74) Agents: MALONEY, Neil, F. et al.; Fenwick & West LLP, Two Palo Alto Square, Palo Alto, CA 94306 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: APPARATUS AND METHOD FOR IMPLEMENTATION OF PPPoE FOR BUS MODEMS



(57) Abstract: An apparatus and method to implement an improved version of point-to-point (PPP) over the Ethernet (PPPoE) for modem communication between a host computer and a server computer. In a first embodiment, the invention provides a method to use PPPoE to connect a host having an operating system to a server through a computer network. In a second embodiment, the invention provides a computer network using PPPoE to connect a host to a server. In a third embodiment, the invention provides a data processing system using PPPoE.

WO 02/19684 A2

Published:
— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Apparatus and Method for Implementation of PPPoE for Bus Modems

Ashok Doddapaneni

Mohammad Moradi

BACKGROUND OF THE INVENTION

1. Related Application

The subject matter of the present application is related to and claims priority from U.S. provisional patent application serial no. 60/230,051, entitled "Implementation of PPPoE for Internal Bus Modems" by Ashok Doddapaneni and Mohammad Moradi, which application was filed on September 1, 2000 and is incorporated herein by reference.

2. Field of the Invention

This invention relates generally to a point-to-point protocol (PPP) implementation over the Ethernet, and, and in particular to an improved PPP implementation over the Ethernet for host computers using USB or PCI modems for communication incorporating a digital subscriber line (DSL) link.

3. Background of the Invention

Digital subscriber line (DSL) is a technology for high-bandwidth connectivity over ordinary copper telephone lines. Recently, broadband network applications are increasingly being implemented on various types of DSL, for example on asymmetric digital subscriber lines (ADSL). There are several DSL technologies; the term "xDSL" refers to the family of DSL technologies, including DSL, ADSL, symmetric DSL (SDSL), high bit rate DSL (HDSL), very high bit rate DSL (VDSL), and rate adaptive DSL (RADSL). A DSL line allows for one line to carry both voice and data signals, and for the data part of the line to be continuously connected.

A DSL modem, located at the phone company's central office, has a POTS splitter, which separates the voice calls from the data. Voice calls are routed to the phone company's public switched telephone network (PSTN) and proceed on their way as usual. Data coming from a PC passes from the DSL modem to a digital subscriber line access multiplexer (DSLAM).

A DSLAM is a network device, usually at a telephone company central office, that receives signals from multiple customer DSL connections and puts the signals on a high-speed

backbone line using multiplexing techniques. Depending on the product, a DSLAM connects a DSL line with some combination of asynchronous transfer mode (ATM), frame relay, or IP networks. A DSLAM enables a phone company to offer business or homes users a fast telephone line technology (e.g., xDSL) with a fast backbone network technology (e.g., an asynchronous transfer mode technology or an equivalent).

FIG. 1 illustrates one example of a prior art implementation of a connection between multiple hosts 102, 104, and 106 to an ISP/corporate server 114. Host 102, 104, and 106 are connected (e.g., through the Ethernet) to CPE 108, which operates as a router to a DSL line to access multiplexer 110 in the central office of the telephone service provider. Access multiplexer 110 is connected through computer network 112 to ISP/corporate server 114.

The DSLAM typically links many DSL lines to a single high-speed asynchronous transfer mode (ATM) line, which in turn connects to the Internet at data transfer speeds up to one gigabyte per second. ATM technology is a high-performance, cell-oriented switching and multiplexing technology that utilizes fixed-length packets to carry different types of traffic. ATM is a technology that enables carriers to capitalize on a number of ATM classes of services; such as high-speed local-area network (LAN) interconnection, voice, video, and future multimedia applications.

FIG. 2 illustrates a more detailed example of a prior art implementation of a connection using a DSLAM and an ATM network between multiple hosts 102, 104, and 106 to an ISP/corporate server 114. Host 102, 104, and 106 are connected (e.g., through the Ethernet) to CPE 108, which operates as a router to a DSL line to DSL access multiplexer 210 in the central office of the telephone service provider. DSL access multiplexer 210 is connected through ATM computer network 212 to ISP/corporate server 114.

ATM wide area, campus, and local area networks are used to transport IP data-grams and other connectionless traffic between hosts, routers, bridges, and other networking devices. The unit of transport in ATM networks is a 53 octet fixed length Protocol Data Unit (PDU) called a cell. A cell consists of a 5-octet header and a 48-octet payload. Variable length PDUs must be segmented by the transmitter to fit into the 48-octet ATM cell payload, and are reassembled by the receiver.

The specification of ATM protocols in request for comment (RFC) 2684 published by the Internet Engineering Task Force (IETF) is the most recent specification, which replaces the specification of ATM protocols in RFC 1483. RFC 2684 describes two encapsulations methods to carry network interconnect traffic over an ATM network. The first method allows multiplexing of multiple protocols over a single ATM virtual connection (VC), whereas the second method assumes that each protocol is carried over a separate ATM VC. RFC 2684 is intended for implementations of ATM networks that carry multi-protocol traffic among hosts, routers, and bridges that are ATM end systems.

Modern access technologies are faced with several conflicting goals. It is desirable to connect multiple hosts at a remote site through the same customer premise access device. It is also a goal to provide access control and billing functionality in a manner similar to dial-up services using the point-to-point protocol (PPP), which provides a standard method for transporting multi-protocol data-grams over point-to-point links. PPP defines an extensible Link Control Protocol and a family of Network Control Protocols (NCPs) for establishing and configuring different network-layer protocols.

In many access technologies, the most cost effective method to attach multiple hosts to the customer premise access equipment, is via the Ethernet. Point-to-point protocol over Ethernet (PPPoE) is a proposal specifying how a host personal computer (PC) interacts with a broadband modem (e.g., xDSL, cable, wireless, and equivalents) to achieve access to the growing number of high-speed data networks. Relying on two widely accepted standards, the Ethernet and the point-to-point protocol (PPP), the PPPoE implementation requires more knowledge on the part of the end user than that required for standard dial-up Internet access. However, PPPoE requires no major changes in the operational model for Internet service providers (ISPs) and carriers.

The significance of PPPoE has to do with its greater ease of use versus competing approaches. PPPoE could speed the widespread adoption of high-speed access services by making high-speed access easier to use for end consumers, and more seamless to integrate into the existing infrastructure for carriers and ISPs. PPPoE also provides a major advantage for a service provider by maximizing integration with, and minimizing disruption of, the existing dial network infrastructures of the ISP. Through tight integration with existing back office automation tools that ISPs have developed for dial customers, PPPoE enables rapid service deployment and cost

savings. PPPoE supports a broad range of existing applications and services, from authentication, accounting, and secure access to configuration management.

The base protocol for PPPoE is described in RFC 2516 published by the Internet Engineering Task Force (IETF). Multiple hosts at a remote site can connect through the same customer premise access device. ISPs are able to provide access control and billing functionality in a manner similar to dial-up services using PPP, and therefore do not need a massive upgrade of their systems.

According to RFC 2516, a host machine seeking access to any network (e.g., the Internet, a corporate network, or an equivalent) uses an Ethernet card and connects to an Ethernet bridging device that also acts as DSL customer premise equipment (CPE). The number of Ethernet ports on the bridge and the terms of agreement with the ISP limit the number of hosts that can connect to this DSL CPE bridge.

PPPoE provides the ability to connect a network of hosts over a simple bridging access device to a remote access concentrator. With this model, each host utilizes its own PPP stack, and the user is presented with a familiar user interface. Access control, billing, and type of service can be done on a per-user, rather than a per-site, basis.

To provide PPPoE, each PPP session must learn the Ethernet address of the remote peer, as well as establish a unique session identifier. PPPoE includes a discovery protocol that provides this. PPP provides a standard method for transporting multi-protocol data-grams over point-to-point links.

The prior art teaches how to build PPP sessions and encapsulate PPP packets over the Ethernet. Prior art facilities are defined for PPP, such as the Link Control Protocol, Network-layer Control Protocols, authentication, and more. These capabilities require a point-to-point relationship between the peers, and are not designed for the multi-point relationships that are available in the Ethernet and other multi-access environments.

The prior art can be used by multiple hosts on a shared Ethernet to open PPP sessions to multiple destinations via one or more bridging modems. The prior art is used with broadband remote access technologies that provide a bridged Ethernet topology, when access providers wish to maintain the session abstraction associated with PPP.

The above description illustrates some of the components of PPPoE technology. What is needed is a simplified network method and apparatus for communication between a host and a server. Moreover, such a method and apparatus preferably should be less expensive than a conventional method and apparatus.

SUMMARY OF THE INVENTION

The present invention provides a simplified network method and apparatus for communication between a host and a server.

The invention provides an improved connection between a host and a server. The invention can be implemented as a method, a system, an apparatus, or in a program on electronically-readable media. Several aspects of the invention are described below.

In accordance with a first aspect of the invention, the invention provides a method to connect a host having an operating system to a server through a computer network. The method includes installing a modem in the host; establishing a connection in the computer network between the host and the server through the modem by using a dial-up service provided by an operating system vendor that supplied the operating system of the host; and communicating information between the host and the server through the computer network.

In accordance with a second aspect of the invention, the invention provides a computer network connecting a host to a server. The computer network includes a host, controlled by an operating system; an xDSL PCI or USB modem that can function as a digital subscriber line customer premise equipment to said host; and a connection from the xDSL PCI or USB modem to a multiplexer, wherein the multiplexer is connected to the server, and the connection between the host and the server includes a dial-up service pro-vided by an operating system vendor that supplied the operating system to control the host.

In accordance with a third aspect of the invention, the invention provides a data processing system. The data processing system includes a server, connected to a network; a host, controlled by an operating system; an xDSL PCI or USB modem that can function as a digital subscriber line customer premise equipment to said host; and a connection from the xDSL PCI or USB modem to a multiplexer, wherein the multiplexer is connected to the server through the network, and the

connection between the xDSL PCI or USB modem and the multiplexer includes a dial-up service provided by an operating system vendor that supplied the operating system to control the host.

These and other objects and advantages of the invention will become apparent to those skilled in the art from the following detailed description of the invention and the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing and other objectives, aspects, and advantages will be better understood from the following detailed description of embodiments of the present invention with reference to the following drawings:

FIG. 1 illustrates one example of a prior art implementation of a connection between multiple hosts to an ISP/corporate server.

FIG. 2 illustrates a more detailed example of a prior art implementation of a connection using a DSLAM and an ATM network between multiple hosts to an ISP/corporate server.

FIG. 3 illustrates one example of the prior art with regards to the components of the host operating system to implement PPPoE.

FIG. 4 illustrates one preferred embodiment of the invention with regards to the components of the host operating system to implement PPPoE.

FIG. 5 illustrates one preferred embodiment of a host using a DSL network interface card (NIC) to communicate by PPPoE to an ISP/corporate server.

FIG. 6 summarizes one preferred embodiment of the invention with regards to the software component protocols used by the host, the DSLAM, and the ISP/corporate server to implement PPPoE.

FIG. 7 illustrates a prior art flow chart for the installation and connection steps required for a customer to implement PPPoE communication.

FIG. 8 illustrates a flow chart for the installation and connection steps required for a customer to implement PPPoE communication according to one preferred embodiment of the invention.

FIG. 9 illustrates a flow chart that shows the sequence of operations that happens in the driver when a packet is sent to the driver from an upper layer (the PPP driver), according to one preferred embodiment.

FIG. 10 illustrates a flow chart that shows the sequence of operations that happens in the driver when a packet is received from the computer network, according to one preferred embodiment.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Alternative embodiments of the invention can be implemented in several environments and applications (e.g., homes, offices, and so forth). The particular environment and application may result in a significantly different configuration than the configurations illustrated below. The advantages of the embodiments of the invention described below also apply to these other environments and applications.

PPPoE has two distinct stages. There is a discovery stage and a PPP session stage. When a Host initiates a PPPoE session, it must first enter the discovery stage to identify the Ethernet Media Access Controller (MAC) address of the peer and establish a PPPoE session identification. While PPP defines a peer-to-peer relationship, the discovery stage is inherently based on a client-server relationship. In the discovery stage, a Host (the client) discovers an Access Concentrator (e.g., a server). Based on the network topology, there may be more than one Access Concentrator with which the Host can communicate. The discovery stage allows the Host to discover all Access Concentrators and then select one. When the discovery stage completes successfully, both the Host and the selected Access Concentrator have the information they will use to build their point-to-point connection over the Ethernet.

The discovery stage remains stateless until a PPP session is established. Once a PPP session is established, both the Host and the Access Concentrator allocate the resources for a PPP virtual interface. When the discovery stage completes, both peers know the PPPoE session identification and the peer's Ethernet address, which together define the PPPoE session uniquely.

There are four steps to the discovery stage. The steps consist of (1) the Host broadcasting an Initiation packet, (2) one or more Access Concentrators sending Offer packets, (3) the Host sending a unicast Session Request packet, and (4) the selected Access Concentrator sending a Confirmation packet. When the Host receives the Confirmation packet, it may proceed to the PPP session stage. When the Access Concentrator sends the Confirmation packet, it may proceed to the PPP session stage.

Once the PPPoE session stage begins, PPP data is sent as in any other PPP encapsulation. All Ethernet packets are unicast. The session identification (Session_ID) does not change for that PPPoE session stage, and is the value assigned in the Discovery stage. The PPPoE payload contains a PPP frame that begins with the PPP Protocol-ID.

PPPoE also uses various software modules for communicating across interfaces between an operating system and various software device drivers. An application programming interface (API) is a set of subroutines or functions that a program, or application, can call to tell the operating system to perform some task. The Windows API consists of more than 1,000 functions that programs written in C, C++, Pascal, and other languages can call to create windows, open files, and perform other essential tasks. For example, an application that wants to display an on-screen message can call Windows' MessageBox API function.

A Network Driver Interface Specification (NDIS) is used to define a standard API for Network Interface Cards (NIC's). The hardware implementation details for a NIC is wrapped by a "Media Access Controller" (MAC) device driver in such a way that all NICs for the same media (e.g., Ethernet media) can be accessed using a common programming interface. NDIS also provides a library of functions (sometimes called a "wrapper") that can be used by MAC drivers as well as higher-level protocol drivers (e.g., TCP/IP). The wrapper functions serve to make development of both the MAC and protocol drivers easier as well as to hide (to some extent) platform dependencies. Corporations such as Microsoft and 3Com Corporation jointly developed early versions of NDIS.

FIG. 3 illustrates one example of the prior art with regards to the components of the host operating system 300 to implement PPPoE. The host operating system 300 includes the TCP/IP module 302 of the original operating system (O/S) vendor, a third party vendor PPPoE module 308, a NDIS module 304 of the original O/S vendor (that supplies an interface to the optional Ethernet module 306), and a RFC 1483 bridge mode module 310 for USB or PCI modems. The Ethernet module 306, if present, connects the NDIS module 304 to the RFC 1483 bridge mode module 310. The Ethernet module 306 includes a device driver that implements the Ethernet protocol and drives an Ethernet NIC. An Ethernet NIC in the host is frequently referred to as an Ethernet port.

When the CPE 108 (shown in FIG. 1) is a router, the connections to the hosts 102, 104, and 106 (shown in FIG. 1) are implemented through the Ethernet. Therefore, the Ethernet Module 306 would be included in the host operating system 300 shown in FIG. 3. However, when the CPE 108 is an xDSL PCI or USB modem, the connections to the hosts 102, 104, and 106 are not implemented through the Ethernet. In this case, the Ethernet Module 306 would not be included in the host operating system 300 shown in FIG. 3.

FIG. 4 illustrates one preferred embodiment of the invention with regards to the components of the host operating system 400 to implement PPPoE. The host operating system 400 includes the TCP/IP module 302 of the original operating system (O/S) vendor, the PPP module 408 of the original O/S vendor, the NDISWAN module 404 of the original O/S vendor, and a PPPoE and RFC 1483 bridge mode module 410 for USB/PCI modems.

More preferred embodiments do not require an Ethernet port in the host computer to implement PPPoE, but alternative embodiments can have one or more Ethernet ports in the host computer. A preferred embodiment without an Ethernet port is in contrast to the prior art, which required at least one Ethernet port in the host computer to implement PPPoE.

FIG. 5 illustrates one preferred embodiment, with a host 104 using an xDSL PCI or USB modem 504 to communicate by PPPoE to an ISP/corporate server 114. Host 104 contains an xDSL PCI/USB modem 504, which is connected to DSL access multiplexer 210 in the central office of the telephone service provider. DSL access multiplexer 210 is connected through ATM computer network 212 to ISP/corporate server 114.

A more preferred embodiment makes use of Microsoft's existing PPP stack and implements PPPoE on top of RFC1483 (or RFC 2684) bridge-mode LLC encapsulation software. To support this, the driver needs to export the NDISWAN interface. Before the first PPP packet (a LCP protocol packet) is sent to the remote PPP server, the discovery stage has to be finished. After the discovery stage is finished, the driver has the MAC address of the Access Concentrator and a Session_ID for the PPP session stage. When the first PPP packet is sent to the driver, the PPPoE header is added with the Access Concentrator's MAC address and the Session_ID before sending packets on the network. The MAC address and Session_ID are used in all traffic, including the PPP connection setup and data transfer.

FIG. 6 summarizes one preferred embodiment of the invention with regards to the software component protocols used by the host 104, the DSLAM 210, and the ISP or corporate server 114 to implement PPPoE. The host 104 uses the following protocols: TCP/IP from the original operating system (O/S) vendor, PPP from the original O/S vendor, PPPoE, Ethernet, RFC 1483 (or RFC 2684) from a third party supplier, ATM, and xDSL. The DSLAM uses the following protocols: ATM and xDSL. The ISP/corporate server 114 uses the following protocols: TCP/IP, PPP, PPPoE, Ethernet, RFC 1483 (or RFC 2684), ATM, and xDSL.

FIG. 7 illustrates a prior art flow chart 700 for the installation and connection steps required for a customer to implement PPPoE communication. The method starts in operation 702. In operation 704 the customer installs a modem in the host computer. Operation 706 is next, where the customer installs the third party PPPoE software in the host computer. Operation 708 is next, where the customer dials up the third party to execute the third party PPPoE software to establish a connection between the host computer and the ISP/corporate server over a computer network. Operation 710 is next, where the customer starts communication. Operation 712 is next, where the customer ends the communication. The method ends in operation 714.

FIG. 8 illustrates a flow chart 800 for the installation and connection steps required for a customer to implement PPPoE communication according to one preferred embodiment. The method starts in operation 802. In operation 804 the customer installs a modem in the host computer. Operation 808 is next, where the customer uses the original O/S vendor software to dial-up the original O/S vendor to establish a connection between the host computer and the ISP/corporate server over a computer network. Operation 810 is next, where the customer starts communication. Operation 812 is next, where the customer ends the communication. The method ends in operation 814.

FIG. 9 illustrates a flow chart 900 that shows the sequence of operations that happens in the driver (i.e., the PPPoE bridge mode module 410 shown in FIG. 4) when a packet is sent to the driver from an upper layer (the PPP driver), according to one preferred embodiment. The method starts in operation 902. In operation 904 a PPP packet is received. Operation 906 is next, where a test is made to determine if a PPPoE session was started. If the test in operation 906 determines no PPPoE session was started, then operation 908 is next. In operation 908 the PPPoE discovery state machine is scheduled to setup a PPPoE session. Operation 910 is next, where a test is made

to determine if a PPPoE session is established. If the test of operation 910 determines a PPPoE session was not established, then operation 908 is repeated. If the test of operation 910 determines a PPPoE session is established, then operation 912 is next. If the test of operation 904 determines a PPP packet was received, then operation 912 is next. In operation 912, PPPoE encapsulation and RFC 1483 bridge mode encapsulation are added and operation 914 is next. In operation 914 the packet is sent over the computer network. The method ends in operation 916.

FIG. 10 illustrates a flow chart 1000 that shows the sequence of operations that happens in the driver (i.e., the PPPoE bridge mode module 410 shown in FIG. 4) when a packet is received from the computer network, according to one preferred embodiment. The method starts in operation 1002. In operation 1004 a PPPoE packet is received. In operation 1006 a test is made to determine if the received packet is of type 0x8863 for the PPPoE discovery stage. If the test in operation 1006 determines that the received packet was type 0x8863, then operation 1008 is next. In operation 1008 the PPPoE discovery state machine is scheduled to setup a PPPoE session, and the packet is passed on. If the test in operation 1006 determines that the received packet was not type 0x8863, then operation 1010 is next. In operation 1010, a test is made to determine if the received packet is of type 0x8864, indicating that a PPPoE session is established. If the test of operation 1010 determines the received packet is not type 0x8864 (a PPPoE session was not established), then operation 1012 is next, where a data error is declared and the packet is dumped. If the test of operation 1010 determines the received packet was type 0x8864 (a PPPoE session is established), then operation 1014 is next. In operation 1014, the RFC 1483 bridge mode encapsulation and PPPoE encapsulation are stripped off. Operation 1016 is next, where the packet is passed to the PPP driver. The method ends in operation 1018.

The exemplary embodiments described herein are for purposes of illustration and are not intended to be limiting. Therefore, those skilled in the art will recognize that other embodiments could be practiced without departing from the scope and spirit of the claims set forth below.

What is claimed is:

1. A method to connect a host having an operating system to a server through a computer network, comprising:

installing a modem in said host;

establishing a connection in said computer network between said host and said server through said modem by using a dial-up service provided by an operating system vendor that supplied said operating system of said host; and

communicating information between said host and said server through said computer network.

2. The method of claim 1, wherein said computer network supports asynchronous transfer mode operations.

3. The method of claim 1, wherein said communicating information between said host and said server routes said information through a digital subscriber line access multiplexer.

4. The method of claim 1, wherein said host supports a protocol chosen from the group of protocols consisting of: TCP/IP, PPP, PPPoE, Ethernet, RFC 1483, RFC 2684, ATM, or xDSL.

5. The method of claim 1, wherein said server supports a protocol chosen from the group of protocols consisting of: TCP/IP, PPP, PPPoE, Ethernet, RFC 1483, RFC 2684, ATM, or xDSL.

6. The method of claim 1, wherein a network driver interface specification wide area network module, and a PPP protocol module are included in said operating system of said host.

7. A computer network connecting a host to a server, comprising:

a host, controlled by an operating system;

a xDSL PCI or USB modem that can function as a digital subscriber line customer premise equipment to said host; and

a connection from said xDSL PCI or USB modem to a multiplexer, wherein said multiplexer is connected to said server, and said connection between said host and said server includes a dial-up service provided by an operating system vendor that supplied said operating system to control said host.

8. The computer network of claim 7, wherein said computer network supports asynchronous transfer mode operations.

9. The computer network of claim 7, wherein said multiplexer is a digital subscriber line access multiplexer.

10. The computer network of claim 7, wherein said host supports a protocol chosen from the group of protocols consisting of: TCP/IP, PPP, PPPoE, Ethernet, RFC 1483, RFC 2684, ATM, or xDSL.

11. The computer network of claim 7, wherein said server supports a protocol chosen from the group of protocols consisting of: TCP/IP, PPP, PPPoE, Ethernet, RFC 1483, RFC 2684, ATM, or xDSL.

12. The computer network of claim 7, further comprising PPPoE client software that is executed on said host to discover one or more access concentrators.

13. The computer network of claim 7, wherein said host includes an ADSL modem.

14. The computer network of claim 7, wherein said host includes a SDSL modem.

15. The computer network of claim 7, wherein said host includes a VDSL modem.

16. A data processing system, comprising:

a server, connected to a network;

a host, controlled by an operating system;

a xDSL PCI or USB modem that can function as digital subscriber line customer premise equipment to said host; and

a connection from said xDSL PCI or USB modem to a multiplexer, wherein said multiplexer is connected to said server through said network, and said connection between said xDSL PCI or USB modem and said multiplexer includes a dial-up service provided by an operating system vendor that supplied said operating system to control said host.

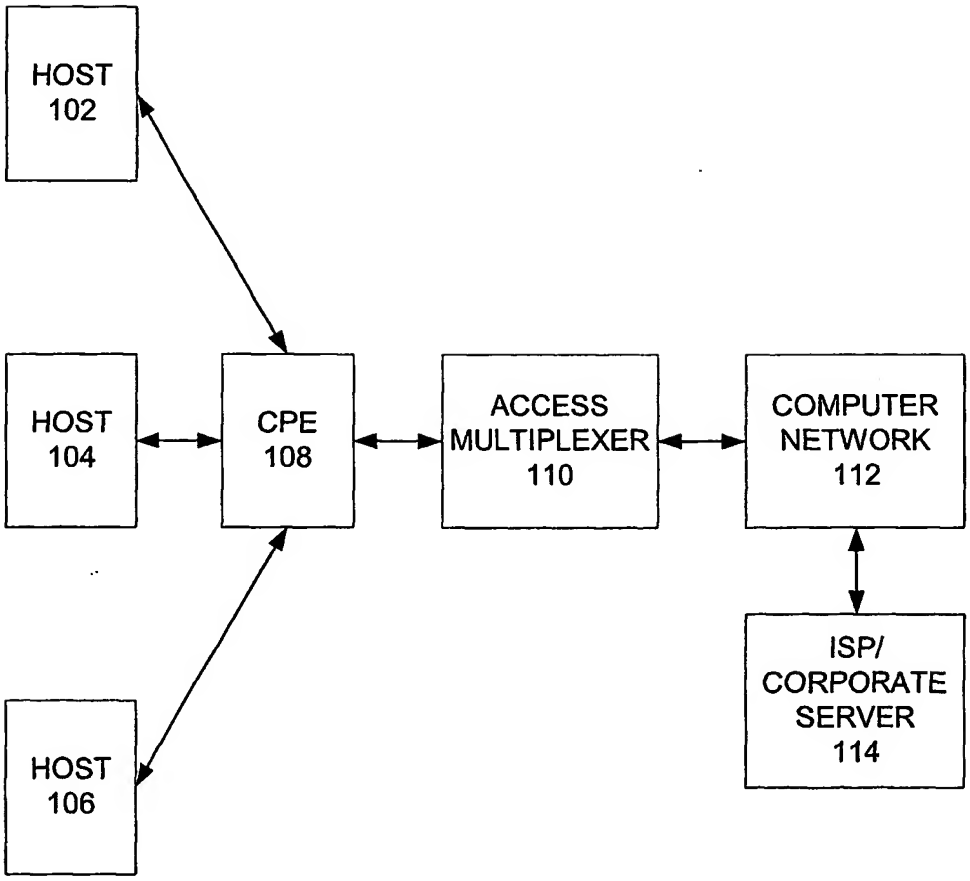
17. The data processing system of claim 16, wherein said network supports asynchronous transfer mode operations.

18. The data processing system of claim 16, wherein said multiplexer is a digital subscriber line access multiplexer.

19. The data processing system of claim 16, wherein said host includes a modem that supports a protocol chosen from the group of protocols consisting of: TCP/IP, PPP, PPPoE, Ethernet, RFC 1483, RFC 2684, ATM, or xDSL.

20. The data processing system of claim 16, wherein said server supports a protocol chosen from the group of protocols consisting of: TCP/IP, PPP, PPPoE, Ethernet, RFC 1483, RFC 2684, ATM, or xDSL.

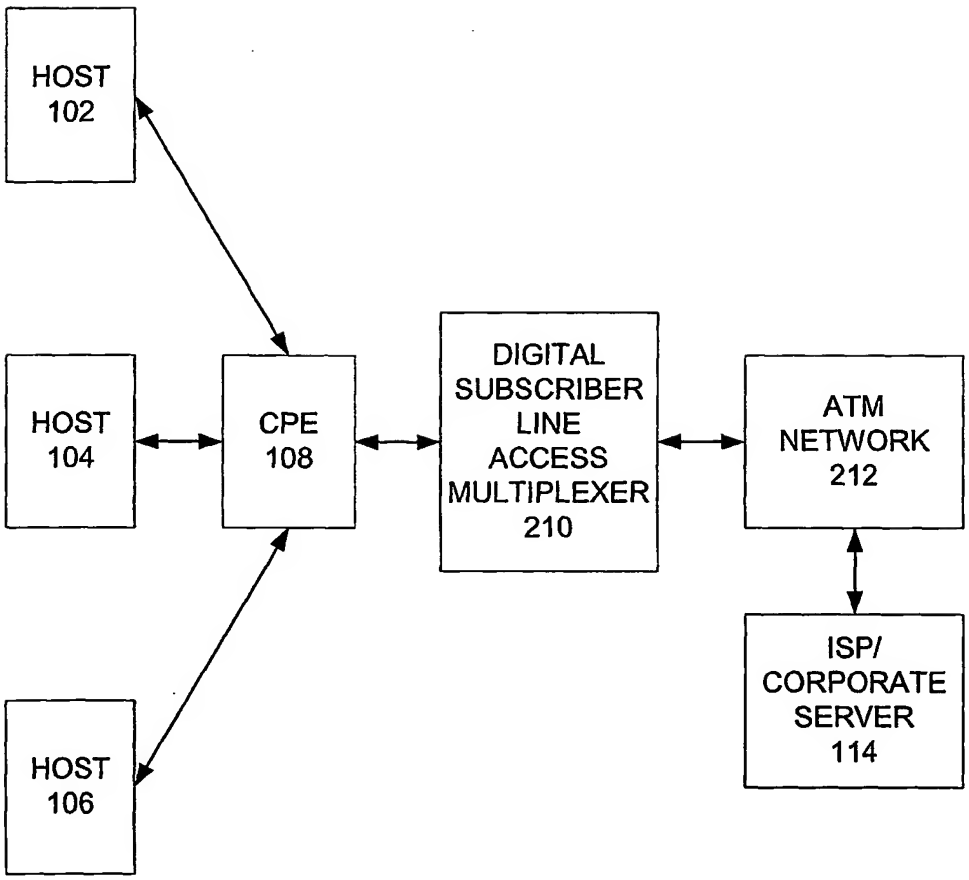
100



(PRIOR ART)

FIG. 1

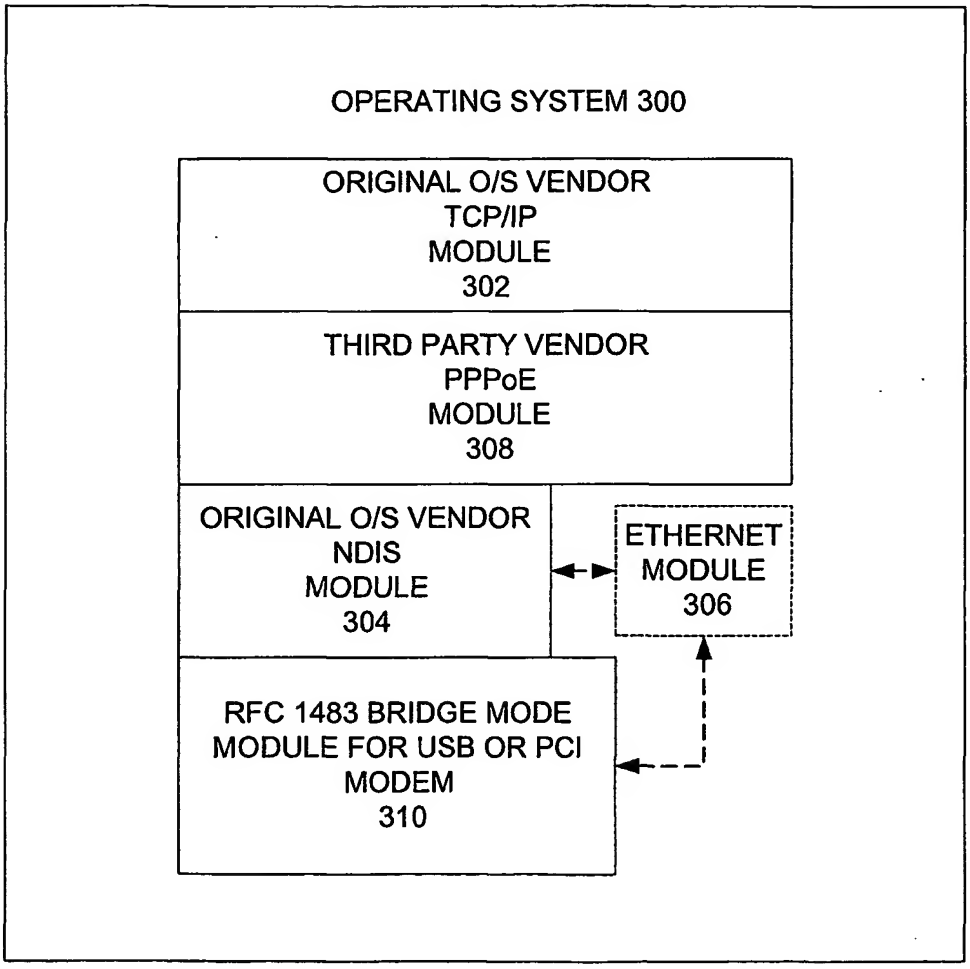
200



(PRIOR ART)

FIG. 2

300



(PRIOR ART)

FIG. 3

400

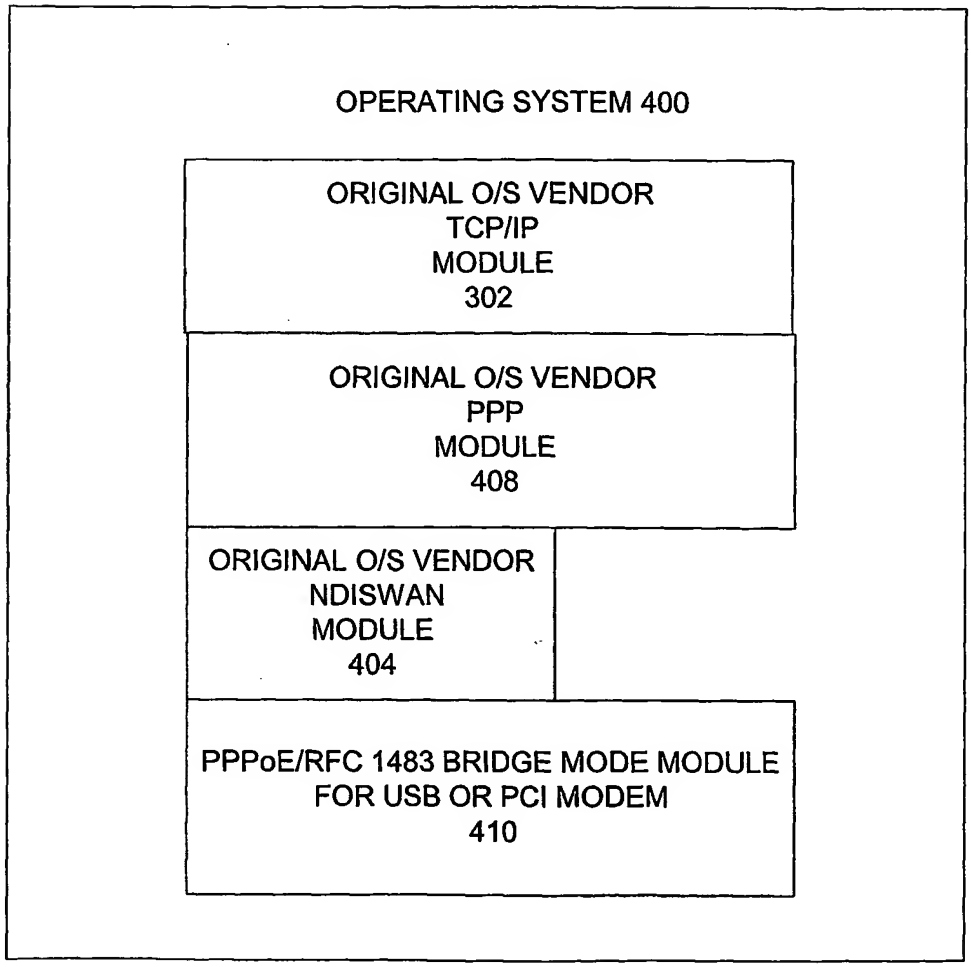

OPERATING SYSTEM 400
ORIGINAL O/S VENDOR TCP/IP MODULE 302
ORIGINAL O/S VENDOR PPP MODULE 408
ORIGINAL O/S VENDOR NDISWAN MODULE 404
PPPoE/RFC 1483 BRIDGE MODE MODULE FOR USB OR PCI MODEM 410

FIG. 4

500

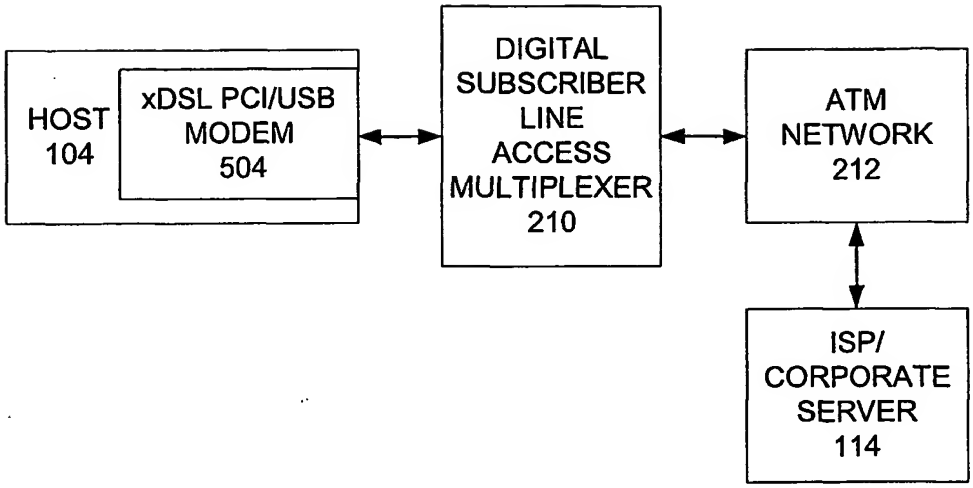

HOST
104
xDSL PCI/USB
MODEM
504
DIGITAL
SUBSCRIBER
LINE
ACCESS
MULTIPLEXER
210
ATM
NETWORK
212
ISP/
CORPORATE
SERVER
114

FIG. 5

600

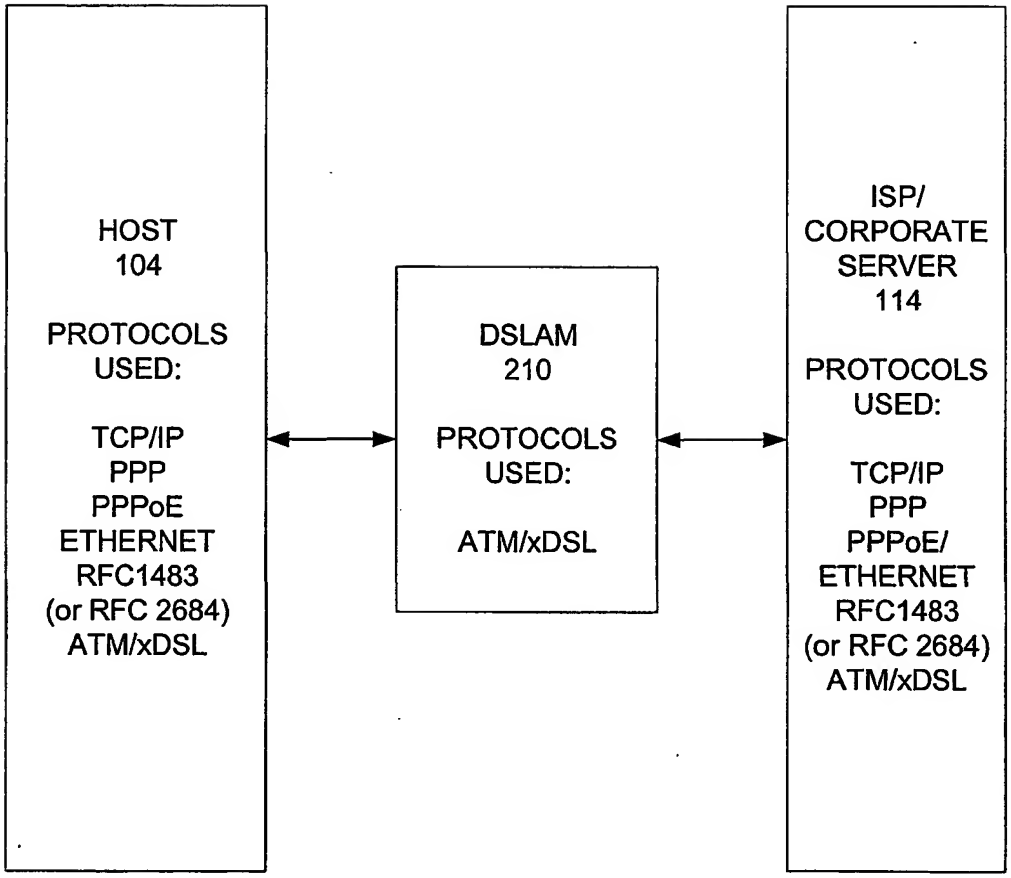
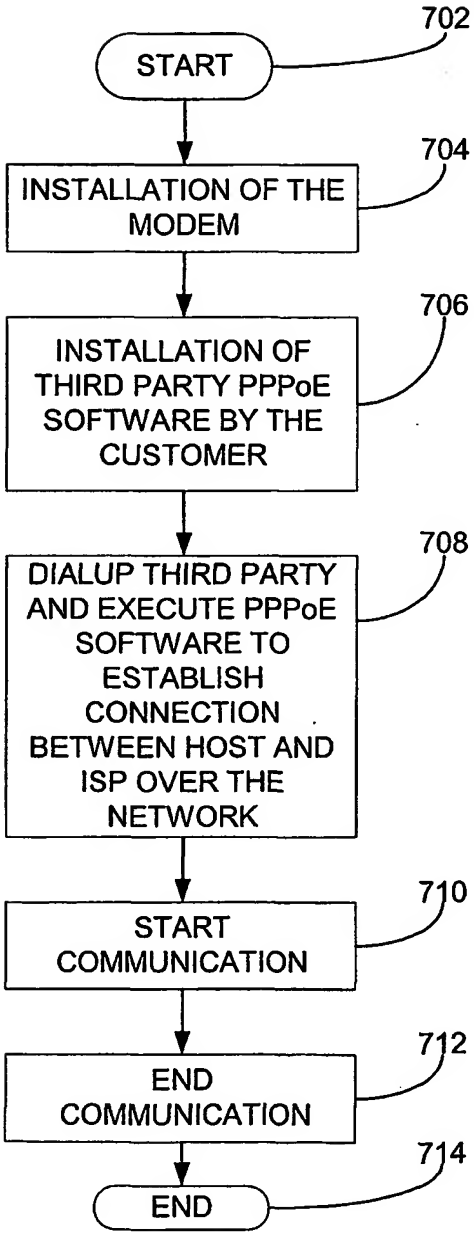

HOST
104
PROTOCOLS
USED:
TCP/IP
PPP
PPPoE
ETHERNET
RFC1483
(or RFC 2684)
ATM/xDSL
DSLAM
210
PROTOCOLS
USED:
ATM/xDSL
ISP/
CORPORATE
SERVER
114
PROTOCOLS
USED:
TCP/IP
PPP
PPPoE/
ETHERNET
RFC1483
(or RFC 2684)
ATM/xDSL

FIG. 6

700



(PRIOR ART)
FIG. 7

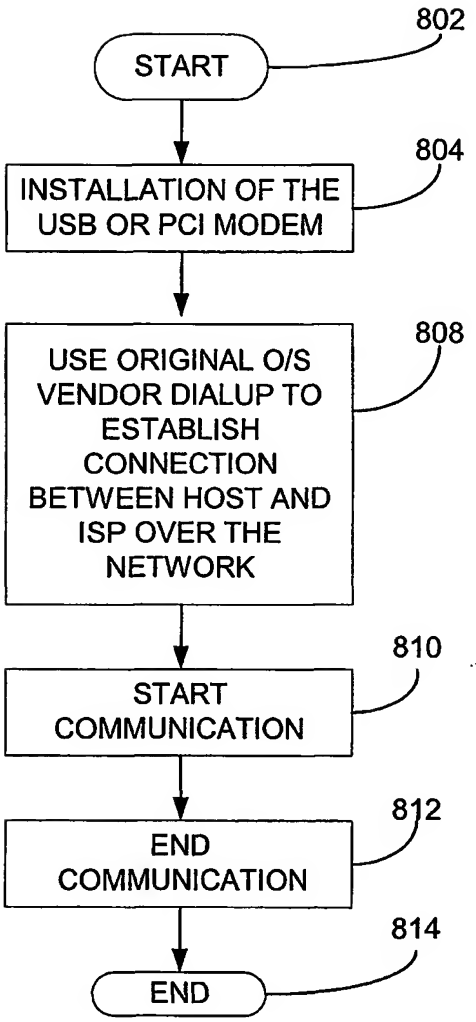

800
802
START
804
INSTALLATION OF THE USB OR PCI MODEM
808
USE ORIGINAL O/S VENDOR DIALUP TO ESTABLISH CONNECTION BETWEEN HOST AND ISP OVER THE NETWORK
810
START COMMUNICATION
812
END COMMUNICATION
814
END

FIG. 8

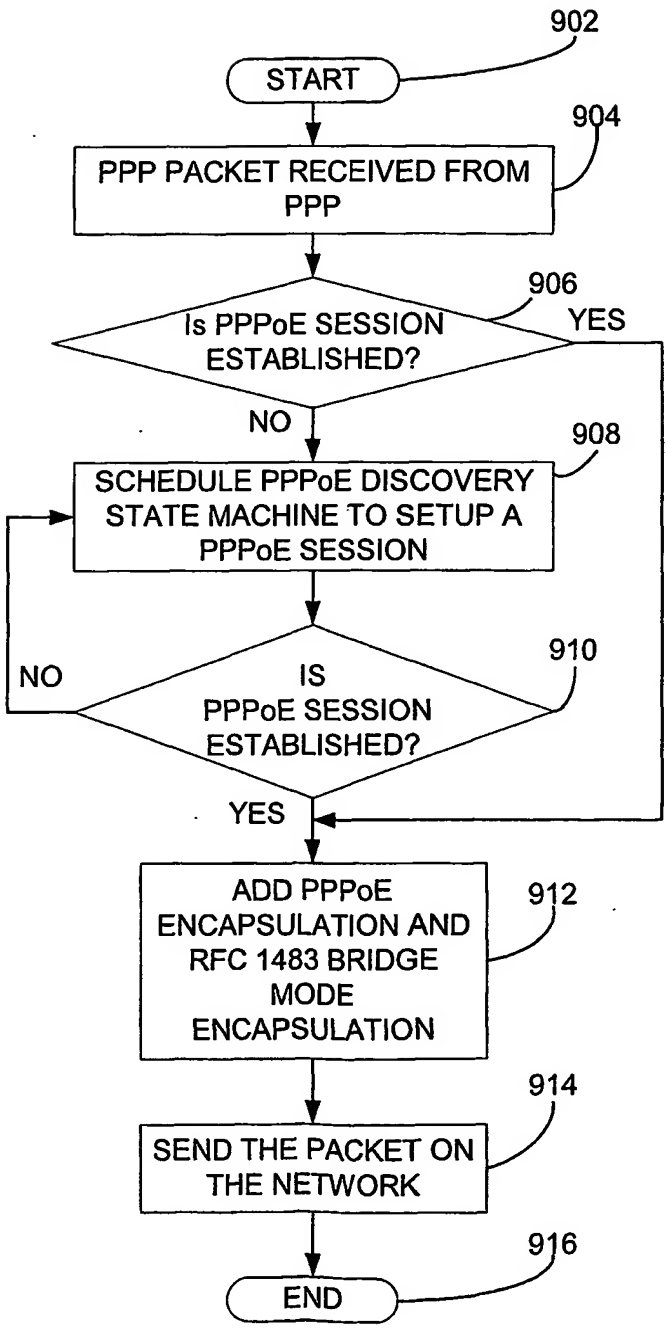

900
902
START
904
PPP PACKET RECEIVED FROM PPP
906
Is PPPoE SESSION ESTABLISHED?
YES
NO
908
SCHEDULE PPPoE DISCOVERY STATE MACHINE TO SETUP A PPPoE SESSION
910
IS PPPoE SESSION ESTABLISHED?
NO
YES
912
ADD PPPoE ENCAPSULATION AND RFC 1483 BRIDGE MODE ENCAPSULATION
914
SEND THE PACKET ON THE NETWORK
916
END

FIG. 9

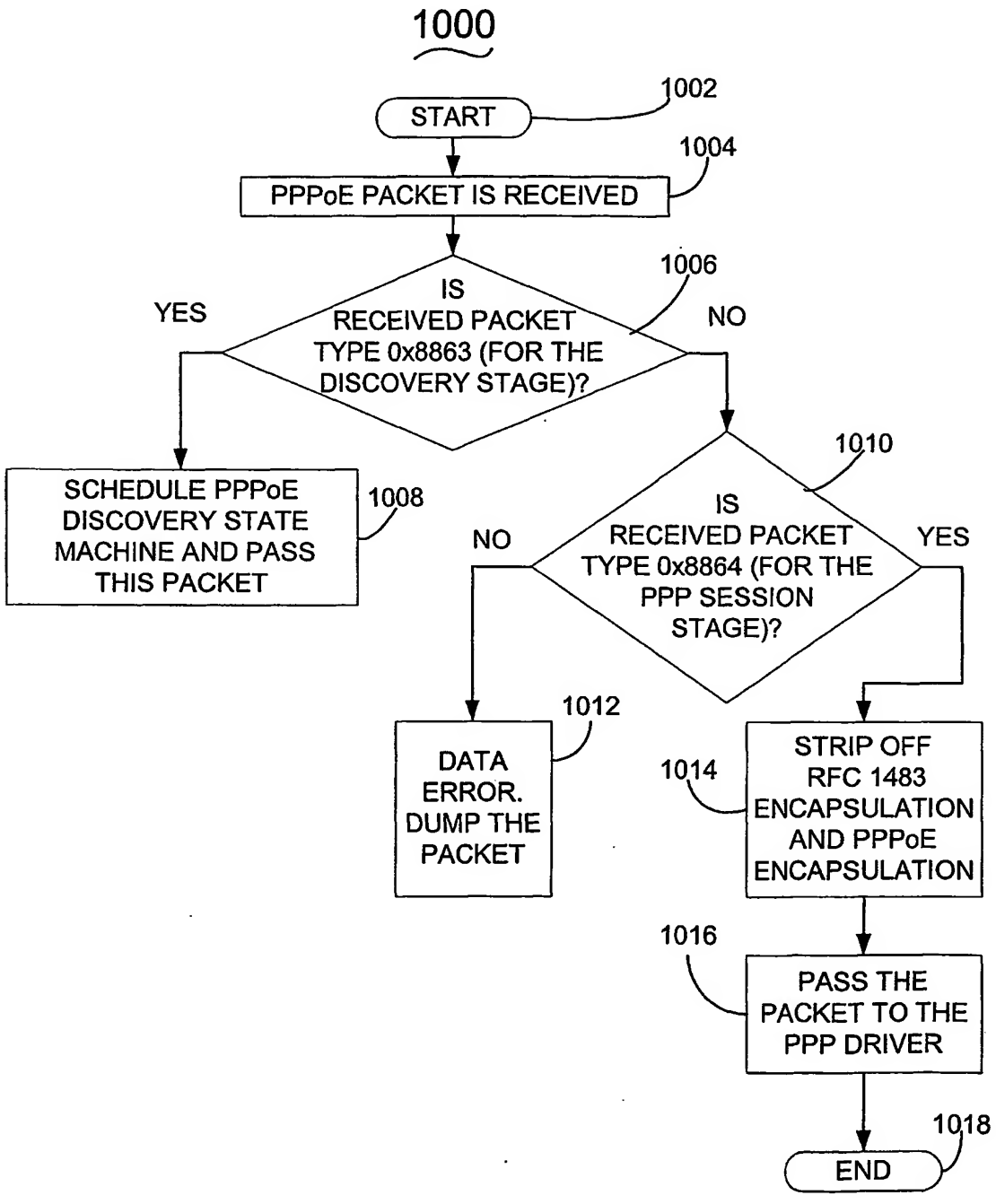

1000
START
1002
PPPoE PACKET IS RECEIVED
1004
IS RECEIVED PACKET TYPE 0x8863 (FOR THE DISCOVERY STAGE)?
1006
YES
NO
SCHEDULE PPPoE DISCOVERY STATE MACHINE AND PASS THIS PACKET
1008
IS RECEIVED PACKET TYPE 0x8864 (FOR THE PPP SESSION STAGE)?
1010
NO
YES
DATA ERROR. DUMP THE PACKET
1012
STRIP OFF RFC 1483 ENCAPSULATION AND PPPoE ENCAPSULATION
1014
PASS THE PACKET TO THE PPP DRIVER
1016
END
1018

FIG. 10